Intelligent In-vehicle Safety and Security Monitoring System with Face Recognition

1st Xiaodi Fu
University of Houston-Clear Lake
Computer Engineering
Houston, TX
fux0339@uhcl.edu

2nd Jiang Lu
University of Houston-Clear Lake
Computer Engineering
Houston, TX
luj@uhcl.edu

3rd Xin Zhang
University of Houston-Clear Lake
Computer Engineering
Houston, TX
zhangx5408@uhcl.edu

4th Xiaokun Yang
University of Houston-Clear Lake
Computer Engineering
Houston, TX
yangxiao@uhcl.edu

5th Ishaq Unwala
University of Houston-Clear Lake
Computer Engineering
Houston, TX
unwala@uhcl.edu

***Abstract*—Dangerous situations such as children are left in vehicles, are dropped off at wrong stops, or take on wrong school buses usually caused by the negligence of drivers. This paper presents a real-time intelligent in-vehicle monitoring system that can count and recognize people as well as alert drivers if such improprieties or potential dangers happen. The system uses HOG-based face detector from Dlib library to obtain face counting function. Face recognition is achieved through two steps, facial feature extraction and face identification. The ResNet is used in facial feature extraction. It transforms an aligned face into a 256-dimensional vector, a Euclidean facial embedding. In face identification, labeled faces will be transformed to facial embeddings first. Then k-nearest neighbor classifier (kNN) is adopted to identify people using such facial embeddings. The simulation on ChokePoint dataset is tested and the average accuracy is 93 percent.**

***Index Terms*—face recognition, feature extraction, ResNet, kNN**

I. Introduction

Nowadays, the security of children on a school bus is an important issue that parents and teachers are concerned about. The issue consists of several aspects. For example, some children are left in school buses, but drivers are not aware of it. In the United States, 37 young children, on average, are killed by vehicular heatstroke each year. Over half of them die merely because they are left in a hot vehicle accident [1] [2]. Also, it is not uncommon that children at junior grades are dropped off at wrong bus stops [3] [4] [5]. To avoid these tragedies, it is significant to monitor children in a school bus and to alert driver when mistakes happen. In the paper, we propose an in-vehicle safety and security system via face recognition. It can help drivers in tracking children when they get on and get off the school bus and alert drivers when dangerous situations occur.

Compared to fingerprint and iris recognition, one of the advantages of face recognition is that it does not have disturbance and does not require cooperation. Although face recognition is widely used in many applications, such as unlocking a smart phone and alerting users when photos of them are posted in Facebook, and its accuracy can reach to 99 percent or higher [6] [7] in some testing benchmarks with given dataset e.g. LFW, it still has many limitations. Variables like illumination, profile faces, occlusion and pose affect accuracy of recognition. Another problem is that most of face recognition models are too big. It requires large amount of computing resources and long time to training or testing that people do not want to install it in mobile devices. In this research, we proposed to build the system that can 1) detect and identify people with high accuracy; 2) run real-time application in edge devices such as raspberry pi; 3) provide feedbacks with notification or warning.

Face detection and recognition are implemented to achieve the functions of the system. Face detection is used to count people in the vehicle and a prerequisite process of face recognition. Popular face detector consists of Haar feature-based cascade detector, Local Binary Patterns (LBP) cascade detector, Histogram of Gradients (HOG) based detector. The first two have API in OpenCV. The last one has API in Dlib. In the research, HOG-based detector is used because it has higher speed compared to the other detectors [8]. Face recognition is a technology of identifying and verifying faces in images or videos by comparing facial features of a face to known faces in a dataset. It consists of face detection, face alignment, feature extraction, and classification. Face alignment is a computer vision method to establish locational correspondences of faces. It tries to warp face images to make key facial points aligned. For example, the center point of two eyes in each face image is in the same spatial location. By aligning faces, the face recognition accuracy can be improved [6]. In this paper, the 68 facial landmarks [9] [10]are used in alignment. Facial feature extraction is the key step to extract features which is used to distinguish faces. Methods like Principal components analysis (PCA) [11], linear discriminant analysis

Fig. 1. Sample frames of real-time face recognition when people get on and off a school bus.

(LDA) [12] and deep neural network, e.g. CNN in FaceNet [6], are popular methods to extract features. Classification algorithms are trained by facial features and corresponding labels. Support vector machines (SVM), kNN and softmax classifier are popular algorithms in classifying faces.

In this paper, we proposed a face recognition method based on ResNet and kNN. The ResNet is introduced for facial feature extraction. The kNN is implemented to classify people using the facial features from ResNet. The rest of the paper is originated as follows. Section II describes the configuration of the system. Face recognition method with architectures are given in section III. Section IV provides the simulation results. Section V concludes the paper.

II. Configuration of the System

Raspberry Pi 3 Model B+, camera module, Wireless adapter, HDMI cable, Ethernet cable, MicroSD card, and power adapter are used in the project. Raspberry Pi 3 Model B+ uses a Linux operating system and provides computing to the project. Camera module offers to capture real-time video. The wireless adapter is used to connect Raspberry Pi to a computer or a mobile phone by a wireless network. Fig. 1 shows that frames of real-time face recognition when people get on and off a school bus.

III. Face Recognition

The procedures of face recognition include face detection, face alignment, face feature extraction, and face identification. Fig. 2 shows the whole framework of face recognition. There are three phases: training in ResNet, training in kNN, and testing in kNN. Details of the face recognition steps and phases are discussed below.

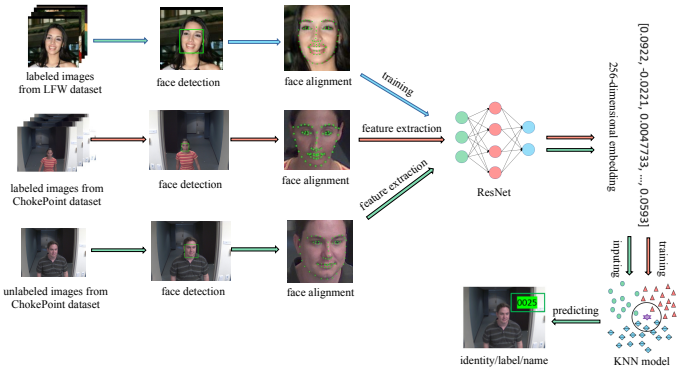

Fig. 2. Framework of Face recognition with ResNet and kNN.

A. Face detection and alignment

The HOG-base face detector in the library of Dlib is implemented for face detection because of its relatively high speed and acceptable accuracy. The face detector in Dlib uses a Maximum-Margin Object Detector [9] with convolutional neural networks (CNN).

Face alignment relies on a pre-trained facial landmark detector [13] in Dlib library. The detector predicts 68 key points that map facial contour. In the paper, we only use those 12 points to align face images. 6 points represent contour of left eye and 6 points represent right eye. We get the coordinate of a center of an eye by coordinates of 6 points. Then we obtain the coordinate of a center of two eyes. The center of two eyes is a center of the rotation in a face image. By rotation, scaling and translation, face images are normalized to implement face recognition [14]. Fig. 3 shows that an original face with an angle of inclination is switched to an aligned face without inclination.

B. Facial feature extraction

A deep neural network based on ResNet is used for facial feature extraction. It is proposed by Kaiming He, Xiangyu Zhang and etc. [15] [16]. It wins the first place in ILSVRC (ImageNet Large Scale Visual Recognition Competition) and COCO (Common Objects in Context) 2015 competition because of good performance in objection classification, detection and segmentation. The advantage of ResNet is ease of training. We use 3 blocks in the ResNet as shown in Fig. 4. One block has 2 convolution layers. Inputs of the ResNet are aligned face images with their labels from one dataset. When an image is put through the ResNet, the output of an

Fig. 3. Face alignment: from original image to a aligned face image.

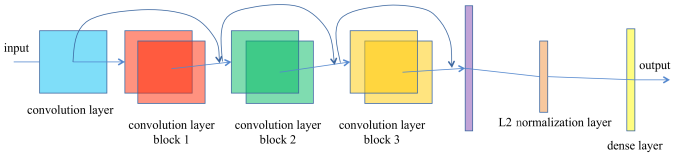

Fig. 4. 3 block resnet architecture.

intermediate layer, L2 normalization layer, is a feature vector of that face. The well-trained ResNet is used as a tool to extract facial features. The feature vector, also called embedding, is used as the input of a kNN model.

C. Face Identification by kNN classifier

We select k-nearest neighbor (kNN) due to its efficiency and robustness to noise when dataset is big. It calculates a distance between the object to be classified to every object in a dataset. Before training, the value of k should be defined. Then, the k neighbors of the object are picked from the dataset. Each neighbor has a vote. A majority vote of the object's neighbors determines which class it is classified. In the paper, we use the output of the embedding layer from ResNet as the input of the kNN model. In this step, labeled images from other dataset are used to train the model. After training the kNN model, unlabeled images also from this dataset are tested. Labels then can be identified.

IV. Results

A. Dataset selection

Two datasets are used in the paper. One is LFW and the other one is Chokepoint. Usually, a network is used as a tool of feature extraction. It is trained on a very large dataset which has from several million to dozens of million images. Several days to weeks are needed to train a model. However, those models often have big size, about 30 layers or more, due to millions of parameters. They may not adaptable in Raspberry Pi or smartphones. We use a ResNet with much fewer layers. LFW has 13233 images and 1680 people with two or more images [17]. Some other deep neural networks, for example, ResNet-34 [15], are used as tools of feature extraction when to recognize several thousand people. Unlike those models, our network is as a tool of feature extraction when recognizing dozens of people. In a school bus, the number of people is not beyond 100, often dozens of. Therefore, we use LFW as our dataset when we train ResNet.

Chokepoint dataset is used to recognize faces. The dataset has 25 people (19 male and 6 female) in part1 and 29 people in part 2 [18]. It is a video dataset which captures different sequence of people when they are walking through and out a portal. It is a similar situation when people get on and get off a vehicle.

B. Facial features with ResNet

In our experiments, we train the ResNet with Adam optimizer with learning rate 0.001 and cross-entropy loss [19]. Kernel initializer is "He normal". A block consists of two

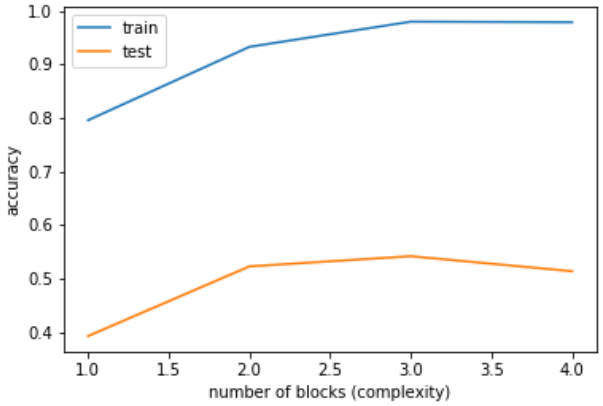

Fig. 5. Complexity of ResNet selection.

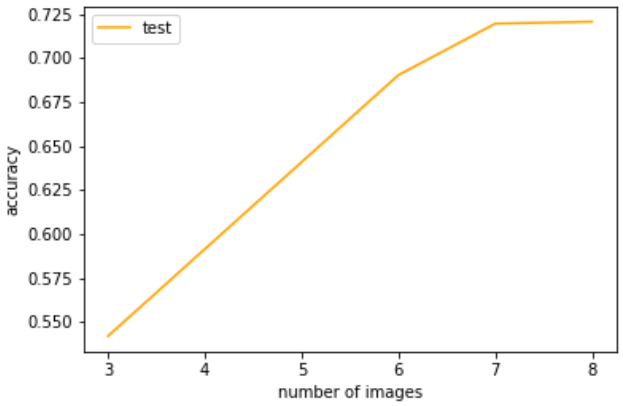

Fig. 6. Testing accuracy with different number of images for one person.

convolution layers and two batch normalization layers. The architecture of the 3 block ResNet is shown in Fig. 4 and the detail of it is shown in Table I. Our model has three blocks which performs better than two blocks and four blocks as shown in Fig. 5. Size of each input image is 224x224 pixels. We train four sub-datasets using the same ResNet. The four sub-datasets are extracted from LFW. They are every people with three or more images, every people with six or more images, every people with 7 or more images and every people with 8 or more images. By comparing different sub-datasets, we found that testing accuracy is better if the dataset has more images for each person as shown in Fig. 6. In order to extract the better model for facial features, we filter people with less than 8 images. So that each person has eight or more than eight images. Additionally, testing accuracy of 72.06 percent is the best when epoch is set to 70. When epoch is bigger than 70, testing accuracy decreases. The loss and accuracy of the ResNet with 3 blocks is shown in Fig. 7 and 8.

In FaceNet, a compact 128-dimensional vector (embedding) represents the key features which distinguish each person. It trained 1 million to 2 million training face images. Davis

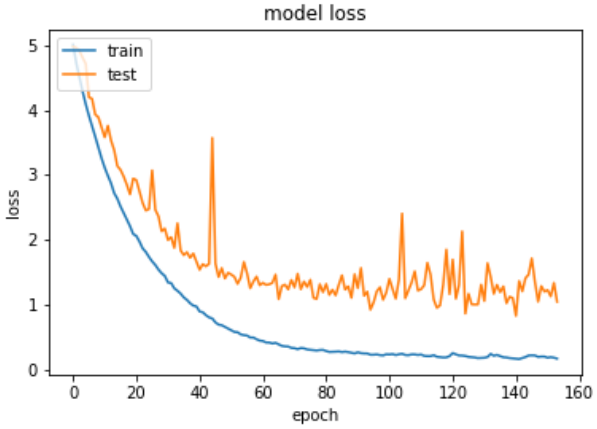

Fig. 7. Loss of the ResNet with 3 blocks.

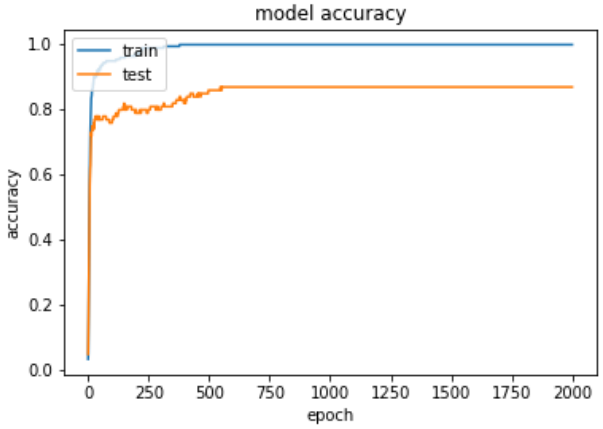

Fig. 9. Accuracy of the softmax classifier.

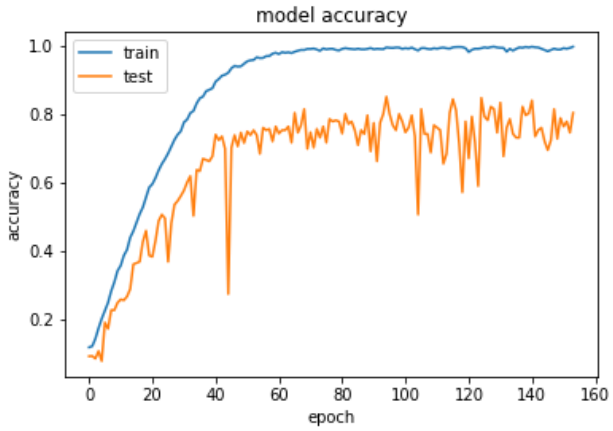

Fig. 8. Accuracy of the ResNet with 3 blocks.

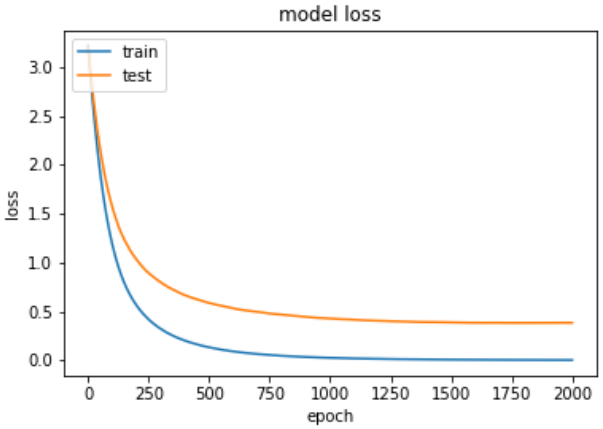

Fig. 10. Loss of the softmax classifier.

King, creator of the Dlib, also use 128-dimensional embedding to represent a face. He trained 3 million face images. However, our dataset of 13233 images is small compared to their dataset. So, we use 256-dimensional embedding to express facial features. The 256-dimensional embedding has more information compared to 128-dimensional embedding. When we use 512-dimensional embedding, the test accuracy is 71.50% which is similar to 256-dimension embedding with test accuracy of 72.06%. Therefore, the 256-dimensional embedding is selected. An image with the size of 224x224x3 is transformed to a 256-dimensional facial feature vector through the ResNet.

C. kNN for identification

In this part, dataset is Chokepoint which is video sequences. We trained the first part of the dataset with 25 persons which means 25 classes. Videos are recorded when those people are walking through a portal one by one. 12 images from every person are extracted randomly from video frames. 8 images are training images and 4 images are used for testing. Therefore, a total of 200 face embeddings are trained in the kNN model. Then, the rest 100 images are used for identification. The root of the number of classes is set as the value of k. Here, k is equal to 5 which means that 5 closest neighbors determines the label of an object. Algorithm used to compute the nearest neighbor is "BallTree". The average accuracy after testing these 100 face embedding is 93%.

D. Softmax classifier for identification

Softmax classifier is often used in soft classification and when dataset has multiple labels or targets. When a sample is predicted by a softmax classifier, it yields probabilities for each class label. Only a label is selected with maximum probability. It is usually the final layer of a neural network. In this paper, it is selected to identify or classify faces. Dataset is the same as the dataset used in kNN for identification. In the classifier, softmax function is used to calculate probabilities. Cross entropy function is used as the loss function, and optimizer is Adam. Relation between accuracy and epochs is shown in Fig. 9, and relation between loss and epochs in shown in Fig. 10. When epoch is bigger than 700, the accuracy and loss

TABLE I
THE STRUCTURE OF THE RESNET

layer name	*input size*	*output size*	*kernel*	*stride*	*parameter*
zero padding	224*224*3	230*230*3	0		0
conv1	230*230*3	112*112*32*3	7*7 32	2	4736
batch normalization1	112*32*3	112*112*32*3	0		128
max pooling1	112*32*3	56*56*32	3*3	2	0
conv2 a	56*56*32	56*56*32	3*3 32	1	9248
batch normalization2 a	56*56*32	56*56*32			128
conv2 b	56*56*32	56*56*32	3*3 32	1	9248
batch normalization2 b	56*56*32	56*56*32			128
add1	56*56*32	56*56*32			0
conv3 a	56*56*32	28*28*64	3*3 64	2	18496
batch normalization 3 a	28*28*64	28*28*64			256
conv3 b	28*28*64	28*28*64	3*3 64	1	36928
conv3 c	56*56*32	28*28*64	3*3 64	2	18496
batch normalization 3 b	28*28*64	28*28*64			256
batch normalization 3 c	28*28*64	28*28*64			256
add2	28*28*64	28*28*64			0
conv4 a	28*28*64	14*14*128	3*3 128	2	73856
batch normalization 4 a	14*14*128	14*14*128			512
conv4 b	14*14*128	14*14*128	3*3 128	1	147584
conv4 c	28*28*64	14*14*128	3*3 128	2	73856
batch normalization 4 b	14*14*128	14*14*128			512
batch normalization 4 c	14*14*128	14*14*128			512
add3	14*14*128	14*14*128			0
max pooling 2	14*14*128	2*2*128	7*7		0
flatten	2*2*128	512			0
dense 1	512	256			131328
normalization embedding	256	256			0
dense softmax	256	184			47288
total parameters					573752

grow very slowly. By training with 2000 epochs, the accuracy of the model reaches 87%.

V. CONCLUSION

We designed an intelligent in-vehicle safety and security monitoring system based on face recognition. We propose a ResNet with relatively small number of parameters (573,752) to extract face features. Some other neural networks in face recognition have more than several million to more than 1 billion. Though they have higher accuracy in face feature extraction, it is hard to apply in mobile device due to a large size. Our ResNet has 3 blocks. Each block has 2 convolution layers. We compared the complexity of resent and proved that 3 blocks model is better than 4 and 2 blocks models. The mechanism consists two phases. One is facial feature extraction. Another is by identifying sequential faces. By experiments of kNN and softmax classifier, kNN has the better accuracy and lower computing complexity. kNN obtains 93 percent accuracy on recognizing 25 people. In our future work, we will focus on increasing accuracy by reducing over-fitting in training, changing architecture of network to adopt the system. Multiple people walking through a portal will also be considered. In real world, it is common that several people get on or off a vehicle at the same time.

REFERENCES

[1] C. Williams and A. Grundstein, "Children forgotten in hot cars: a mental models approach for improving public health messaging," Injury Prevention, vol. 24, (4), pp. 279-287, 2018.

[2] The Noheatstroke Website.[Online Apr. 2019]. Available. https://www.noheatstroke.org/original/

[3] Anonymous "4-year-old dropped off at wrong bus stop on first day of school," The Washington Post (Online), 2016.

[4] "Spruce Grove girl, 5, dropped off at wrong bus stop; left alone in the cold," The Canadian Broadcasting Corporation (CBC), 2016.

[5] V. S. Martin, "Dropped off at the wrong bus stop? Find out what this pre-K student did next: The 4-year-old Prince George's Co. girl crossed a street with the help of a Good Samaritan," The Washington Post (Online), 2016.

[6] F. Schroff, D. Kalenichenko and J. Philbin, " FaceNet: A Unified Embedding for Face Recognition and Clustering," IEEE Conf. Proc. on computer vision and pattern recognition, pp. 815-823. 2015.

[7] The Dlib Website.[Online Apr.2019]. Available. http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html

[8] F. Chang et al, "FacePoseNet: Making a case for landmark-free face alignment," In Proceedings of the IEEE Conf. on Computer Vision, pp. 1599-1608. 2017.

[9] Kazemi V, Sullivan J. "One millisecond face alignment with an ensemble of regression trees"In IEEE Conf. Proc. on computer vision and pattern recognition, pp. 1867-1874. 2014.

[10] H. Wang, J. Lu and T. Zhang, "eCamera: A Real-time Facial Expression Recognition System," Proceedings on the International Conference on Artificial Intelligence (ICAI), pp. 264-270, 2018

[11] M. TURK and A. PENTLAND, "EIGENFACES FOR RECOGNITION," Journal of Cognitive Neuroscience, vol. 3, (1), pp. 71-86, 1991.

[12] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, (7), pp. 711-720, 1997.

[13] C. Sagonas et al, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in 2013, . DOI: 10.1109/ICCVW.2013.59.

[14] A. Rosebrock. "Facial landmarks with dlib, OpenCV, and Python". [Online Apr. 2017].Available:https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/

[15] K. He et al, "Deep residual learning for image recognition," In IEEE Proc. on computer vision and pattern recognition, pp. 770-778. 2016.

[16] K. He et al, "Identity Mappings in Deep Residual Networks," 2016.

[17] The Labeled Faces in the Wild Website.[Online Apr. 2019]. Available. http:// vis-www.cs.umass.edu/lfw/

[18] Y. Wong et al, "Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition," In CVPR 2011 WORKSHOPS, pp. 74-81. IEEE, 2011.

[19] The facenet Website.[Online Apr. 2019]. Available. https://github.com/davidsandberg/facenet